

Aula Prática: Tipo Abstrato de Dados

Objetivo: Capacitar o aluno a trabalhar com especificação e implementação independente (em arquivos separados) de Tipos Abstratos de Dados (TAD) simples.

Exercícios:

1. Especifique e implemente um TAD que represente números complexos. Sabe-se que um número complexo é escrito da forma **$a + bi$** , sendo **a** a sua parte real e **b** a sua parte imaginária. Ambas as partes representam valores reais e podem ser zero. Esse TAD deve contemplar as seguintes operações:

- (a) **cria**: cria um nº complexo sem valor atribuído.
- (b) **set_nro**: atribui os valores das partes real e imaginária de um nº complexo.
- (c) **get_nro**: obtém os valores das partes real e imaginária de um nº complexo.
- (d) **apaga**: elimina um número complexo.
- (e) **soma**: soma dois números complexos.
- (f) **sub**: subtrai dois números complexos.
- (g) **multi**: multiplica dois números complexos (lembrando que $i^2 = -1$)

O programa aplicativo irá ler dois números complexos (C1 e C2) e apresentar o resultado das seguintes operações: (C1 + C2); (C1 – C2); e (C1 * C2). Para isto, o programa deve solicitar ao usuário que informe a parte real e imaginária dos dois números complexos e criá-los (um de cada vez). Em seguida, realizar as operações solicitadas e apresentar o resultado na tela. Por fim, os números complexos devem ser apagados antes do programa ser encerrado.

OBS: se necessário, pesquisar como fazer as operações com N^{os} complexos.

2. Especifique e implemente o TAD ponto para representar pontos no espaço tridimensional (R^3). Esse TAD deve contemplar as seguintes operações:

- (a) **cria_pto**: gera um ponto R^3 sem os valores das coordenadas.
- (b) **set_pto**: atribui os valores das coordenadas de um ponto.
- (c) **get_pto**: obtém os valores das coordenadas de um ponto.
- (d) **apaga_pto**: elimina um ponto existente.

(e) **dist**: calcula a distância entre dois pontos R^3 .

O programa aplicativo deve ler as coordenadas de 2 pontos (digitadas pelo usuário) e mostrar a distância entre esses pontos. Para isto, este aplicativo deverá criar os dois pontos, calcular e mostrar a distância entre eles e depois apagá-los.

OBS: se necessário, pesquisar como calcular a distância entre 2 pontos R^3 .

3. Utilizando o TAD ponto criado no exercício anterior, especifique e implemente o TAD esfera. Vale destacar que, uma esfera pode ser representada pelo seu raio, o qual é uma reta entre 2 pontos. Esse TAD deve contemplar as seguintes operações:

- (a) **cria_esfera**: gera uma esfera sem os pontos definidos.
- (b) **set_esfera**: atribui os pontos que determinam o raio da esfera.
- (c) **get_esfera**: obtém os pontos do raio de uma esfera.
- (d) **apaga_esfera**: elimina uma esfera existente e seus respectivos pontos.
- (e) **raio**: determina o comprimento do raio da esfera.
- (f) **área**: determina a área da esfera.
- (g) **volume**: determina o volume da esfera.

O programa aplicativo deve ler os dados da esfera (pontos que determinam o seu raio) e apresentar o comprimento do seu raio, sua área e seu volume. Ao final, a esfera deve ser apagada.

OBS: o programa aplicativo deve usar os TADs ponto R^3 e esfera que, por sua vez, também usará o TAD ponto R^3 (lembre-se de incluir as respectivas diretivas `#include`). Caso necessário, pesquisar como se calcula o raio, área e volume de uma esfera.